**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет   
імені Петра Могили**

**Факультет комп’ютерних наук**

**Кафедра інтелектуальних інформаційних систем**

**ЗВІТ**

**З ПРАКТИЧНОЇ РОБОТИ № 2**

з дисципліни «Розробка web-застосунків на базі ASP.NET»

на тему: **«Впровадження залежностей у компоненти Middleware»»**

Спеціальність 122 «Комп'ютерні науки»

122 – ЛР.ПЗ.03 – 401.22010127

Виконав: студент групи 401

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_О. В. Фоменко

(підпис, ініціали та прізвище)

19.02.2024

(дата)

Перевірила: викл. каф. комп. інж.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Є.Д. Стоев

(підпис, ініціали та прізвище)

(дата)

**Миколаїв – 2024**

**Практична робота №3. «Впровадження залежностей у компоненти Middleware»**

**Завдання**

1. Створіть клас із методами для арифметичних операцій з іменем CalcService. Створіть контролер, який міститиме методи дії для арифметичних операцій. Налаштуйте DI таким чином, щоб контролер користувався сервісом CalcService.

2. Створіть невеликий сервіс із життєвим циклом Transient. Сервіс повинен аналізувати поточний час на сервері і, залежно від результату, у формі http відповіді повертати рядок «зараз день» (якщо час між 12 та 18 годинами), «зараз вечір» (якщо між 18 та опівночі), «зараз ніч» (від півночі до 6 ранку) і, відповідно, «зараз ранок» (від 6 до 12).

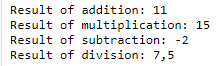
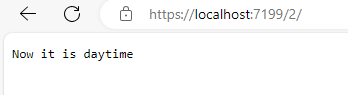
 

Рисунок 1 – Результат роботи застосунку

Посилання на GitHub з кодом застосунку:

[OleksiyFomenko/ASP.NET\_University at LR3 (github.com)](https://github.com/OleksiyFomenko/ASP.NET_University/tree/LR3)

**Контрольні питання**

1. **Як відбувається реєстрація сервісів у ASP.NET Core?**

Існує кілька способів здійснення реєстрації сервісів у ASP.NET Core.

По-перше, в рамках вбудованих методів, ASP.NET Core вже має певні вбудовані функціональності для реєстрації популярних сервісів, таких як IHttpContextAccessor, ILoggerFactory і інші.

Другий метод використовує методи розширення, де для реєстрації груп пов'язаних сервісів використовується метод Add{GROUP\_NAME}.

Крім того, існує можливість прямої реєстрації, де ви можете самостійно зареєструвати сервіси, використовуючи IServiceCollection та методи AddSingleton, AddScoped або AddTransient.

1. **Які варіанти життєвого циклу сервісу?**

Життєвий цикл сервісу визначається областю видимості та типом реєстрації. Основні варіанти циклу життя служб у ASP.NET Core:

* Transient: Кожен раз, коли запитується сервіс, створюється новий екземпляр.
* Scoped: Створюється один екземпляр для кожного запиту (запитання користувача) у межах області видимості (зазвичай, це один HTTP-запит).
* Singleton: Створюється єдиний екземпляр на весь період роботи застосунку.

1. **Що, буде якщо на один інтерфейс у застосунку буде зареєстровано кілька реалізацій сервера?**

Якщо на один інтерфейс у застосунку буде зареєстровано кілька реалізацій сервісу, ASP.NET Core вибере реалізацію, яка визначається останньою реєстрацією. Іншими словами, останній виклик методу реєстрації перезаписує попередні.